Les petits plats

Fiche d'investigation de fonctionnalité

Fonctionnalité : Recherche	**Fonctionnalité #3**
Problématique : Rechercher la méthode la plus performance pour implémenter une fonction de recherche "temps réel" suite à une saisie de l'utilisateur. celle-ci doit démarrer dès le 3eme caractère tapé et afficher les recettes contenant les caractères saisis. La recherche s'effectue sur les champs titre, description et ingrédients	

Option 1 : Utilisation des fonctions natives de manipulation de tableaux Javascript (map, select, reduce, …)
Dans cette option, nous implémentons l'algo avec les fonctions natives censées être plus rapides.

- Enumération des recettes avec la fonction map
- Énumération des ingrédients avec la fonction 'reduce'

Avantages	**Inconvénients**
⊕ L'algo est très compact ⊕ Utilisation des fonctions optimisées de JS	⊖ Perf dépendante de l'implémentation du navigateur ⊖ Lisibilité immédiate (reduce, filter)

Mesures de performance:

- Utilisation de JSBen.ch : voir la page https://jsben.ch/6j1NO
- Test "in-situ" des deux fonctionnalités sur la branche git "mesure-des-deux-fonctionalites", résultats dans la console

Option 2 : Utilisation des boucles "legacy"
Dans cette option nous n'utilisons que les fonctions de base du Javascript pour faire la recherche

- Boucle for pour énumérer la liste des recettes
- Boucle for pour énumérer les ingrédients

Avantages	**Inconvénients**
⊕ Code plus lisible ? ⊕ Perf peu dépendante de l'implémentation	⊖ Potentiellement moins performant

Mesures de performance:

- Utilisation de JSBen.ch : voir la page https://jsben.ch/6j1NO
- Test "in-situ" des deux fonctionnalités sur la branche git "mesure-des-deux-fonctionalites", résultats dans la console

Conclusions:

JSBen.ch et le test in-situ montrent que les performances des deux solutions sont très proches. Dans le pire cas, la moins performante des deux solutions prend entre 70 et 200 microsecondes pour faire la recherche, ce qui est négligeable en tant que temps de réponse utilisateur.

La solution utilisant les fonctions natives de manipulation de tableaux donne un code plus compact et utilise des fonctions natives du Javascript

Solution retenue : Option 1 - Utilisation des fonctions natives de manipulation de tableaux Javascript

Annexes

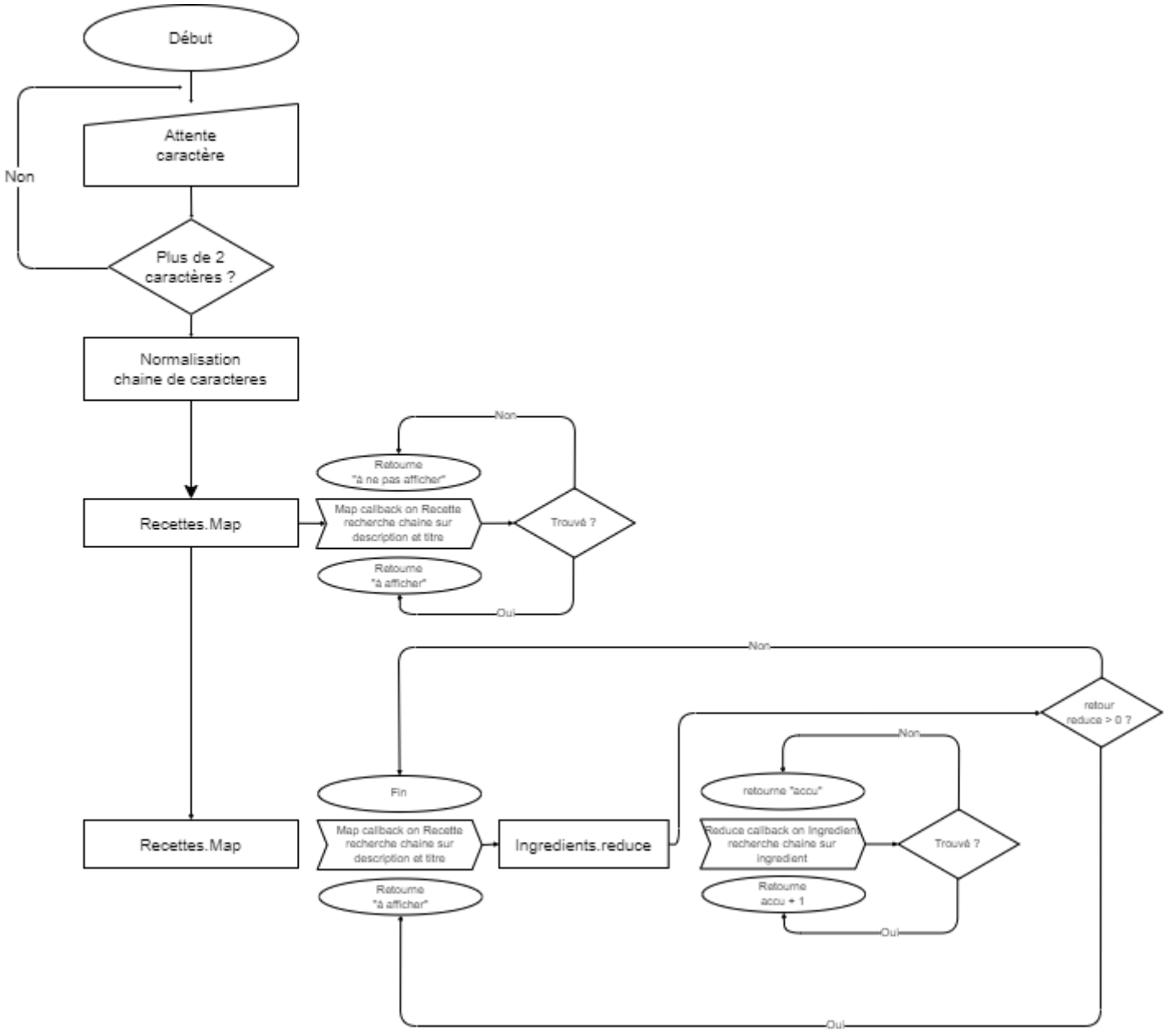

Figure 1 - Implémentation “legacy”

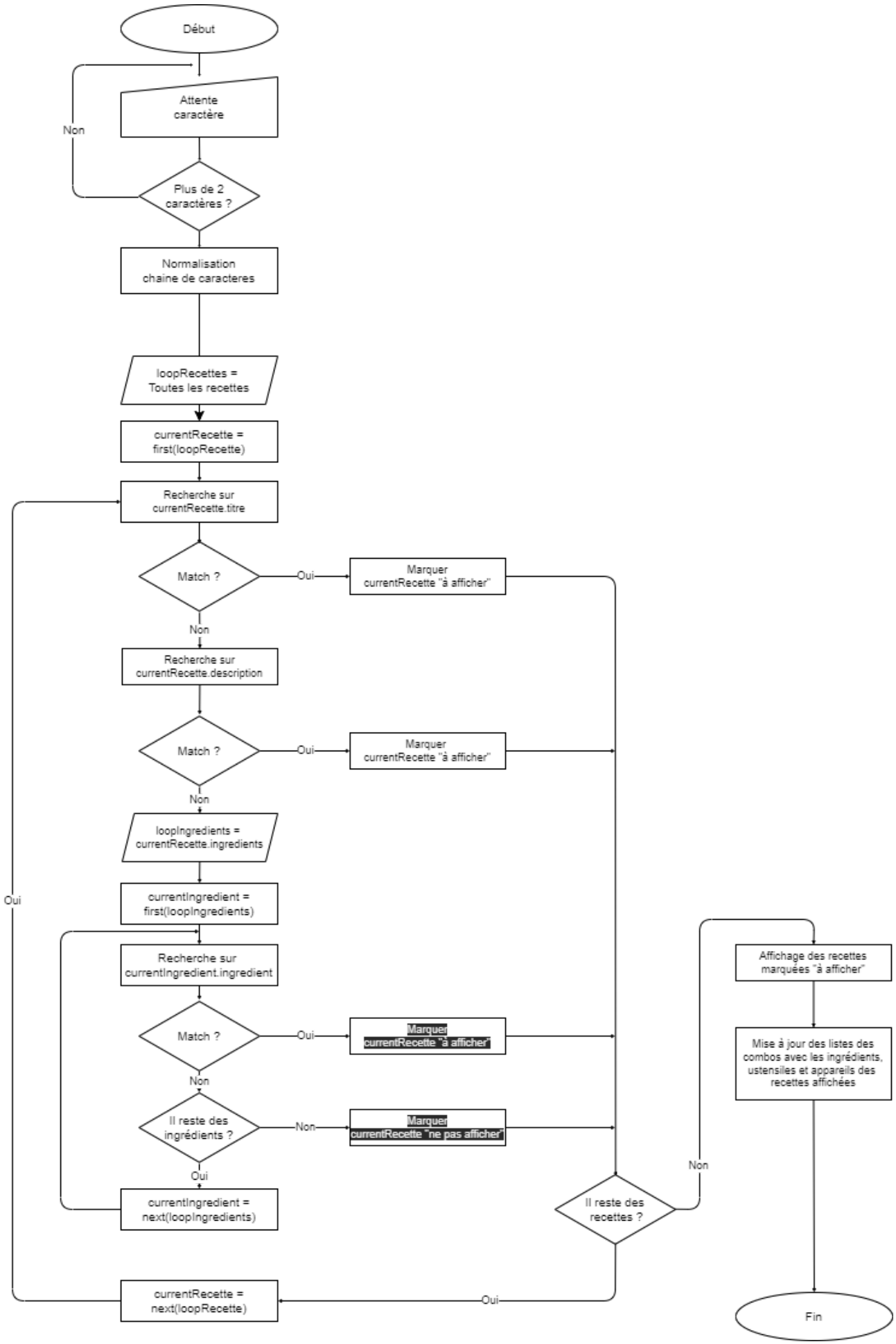

Figure 2 : Implémentation “array method”